

Listing of Claims

1. (Currently Amended) An apparatus ~~comprising storing a computer-readable module for use in a host computer, where the module comprises:~~
a computer-readable module for use in a host computer, wherein the module comprises:
 - a functional component for providing a predetermined programming functionality;
 - an installation component that manages installation of the functional component, wherein
the installation component comprises logic configured to install the functional component on the host computer; and
 - a preprocessing interface for providing introspective information about any requirements of the functional component in response to a request from the host computer.
2. (Original) The apparatus of claim 1 wherein the preprocessing interface comprises a command sequence that identifies the functional component.
3. (Original) The apparatus of claim 1 wherein the preprocessing interface comprises executable object code that identifies the functional component.
4. (Original) The apparatus of claim 1 wherein the request from the host computer occurs as part of an installation of the module in the host computer.
5. (Original) The apparatus of claim 1 wherein the request from the host computer occurs when the module is required by a second module stored at the host computer.
6. (Previously Presented) The apparatus of claim 1 wherein the introspective information provided by the preprocessing interface comprises version information for the module.

7. (Previously Presented) The apparatus of claim 1 wherein the introspective information provided by the preprocessing interface comprises identification of an external module required by the functional component.

8. (Previously Presented) The apparatus of claim 1 wherein the introspective information provided by the preprocessing interface comprises identification of the type of functional component.

9. (Previously Presented) The apparatus of claim 1 wherein the introspective information provided by the preprocessing interface comprises identification of at least a first resource at the host computer that will be modified by the functional component.

10. (Currently Amended) A computer system comprising a first computer, where the first computer comprises:

a processor; and

a memory, wherein

the memory is coupled to the processor, and

the memory comprises:

a plurality of plugin modules, including at least a first plugin module comprised of an introspection interface, an installation program component and a program behavior portion, wherein
the installation program component comprises logic configured to
install the program behavior portion on the first computer;

and

a harness for evaluating the introspection interface of the first plugin module to determine if any resources are required for use of the program behavior portion of the first plugin module.

11. (Original) The computer system of claim 10 further comprising a second computer connected to the first computer, where the second computer includes a first resource required by the first plugin module that is transferred to the first computer upon request by the plugin installation program component.

12. (Original) The computer system of claim 10 further comprising a second computer connected to the first computer, where the second computer includes a first resource required by the first plugin module that is transferred to the first computer upon request by the harness.

13. (Currently Amended) The computer system of claim ~~11~~12, wherein the first resource comprises a second plugin module comprised of an introspection interface and a program behavior portion, and wherein the harness is further structured to evaluating the introspection interface of the second plugin module to determine if any resources are required for use of the program behavior portion of the second plugin module.

14. (Previously Presented) The computer system of claim 10 wherein the harness comprises:

- a loader component for loading a plugin module comprised of an introspection interface and a program behavior portion;
- a validator component for interfacing with the introspection interface to identify any resources required by the plugin module; and
- a finder component for surveying the computer system for any resources identified by the validator component.

15. (Previously Presented) The computer system of claim 10 wherein the introspection interface is an executable script, command series or object-code module.

16. (Previously Presented) The computer system of claim 10 wherein the installation manager is an executable script, command series or object-code module.

17. (Original) The computer system of claim 10 wherein the computer system is a client/server system and the first plugin module is a client plugin module.

18. (Original) The computer system of claim 10 wherein the computer system is a client/server system and the first plugin module is a server plugin module.

19. (Currently Amended) A method of ~~preprocessing a software module~~ comprised of an interface and a program portion, comprising:
receiving a first software module, wherein
the first software module comprises
an interface,
a program portion, and
an installation manager portion;
querying the interface of the first software module to identify any resources
required by the program portion; and
installing the first software module if all resources required by the program
portion are available, wherein
the installing is performed by the installation manager portion of the first
software module.

20. (Previously Presented) The method of claim 19, further comprising
retrieving any resources required by the program portion that were identified as a result
of querying the interface prior to invocation of the first software module.

21. (Currently Amended) The method of claim 19, further comprising
retrieving any resources required by the program portion that were identified as a result
of invoking ~~an~~ the installation manager portion of the first software module.

22. (Previously Presented) The method of claim 20 wherein the retrieved
resource comprises a second software module comprised of an introspective interface and
a program portion, further comprising querying the interface of the second software
module to identify any resources required by the program portion of the second software
module.

23. (Original) The method of claim 19, wherein the step of receiving a first
software module comprises connecting to a remote computer over a telecommunication
network and downloading the first software module from the remote computer.

24. (New) The apparatus of claim 1, wherein the module comprises a new version of the functional component, a previous version of the functional component has been installed on the host computer, and the installation component installs the new version of the functional component on the host computer.
25. (New) The apparatus of claim 24, wherein the new version of the functional component does not require a change to a module on the host computer.
26. (New) The apparatus of claim 1, wherein the module is self-installing such that a change to a second module on the host computer is not needed to implement a change to an installation of the functional component on the host computer.
27. (New) The computer system of claim 10, wherein the first plug-in module comprises a new version of the program behavior portion, a previous version of the program behavior portion has been installed on the host computer, and the installation component installs the new version of the program behavior portion on the host computer.
28. (New) The computer system of claim 27, wherein the new version of the program behavior portion does not require a change to a module on the host computer.
29. (New) The computer system of claim 10, wherein the first plug-in module is self-installing such that

a change to a second module on the host computer is not needed to implement a change to an installation of the program behavior portion on the host computer.

30. (New) The method of claim 19, wherein the first software module comprises a new version of the program portion, a previous version of the program portion has been installed on the host computer, and the installation component installs the new version of the program portion on the host computer.

31. (New) The method of claim 30, wherein the new version of the program portion does not require a change to a module on the host computer.

32. (New) The method of claim 19, wherein the first software module is self-installing such that a change to a second module on the host computer is not needed to implement a change to an installation of the program portion on the host computer.